Ancient handwritings decomposition into graphemes and codebook generation based on Graph coloring

H. Daher[2], D.Gaceb[2], V. Eglin[2], S. Bres[2], N. Vincent[1]
[1.] *Université René Descartes-CRIP5- Systèmes Intelligents de Perception - 75270 Paris 2 LIRIS*
[2.] *Université de Lyon, CNRS, INSA-Lyon, LIRIS, UMR5205 – 69621*
{hani.daher, djamel.gaceb1,veronique.eglin, stephane.bres} @insa-lyon.fr
nicole.vincent@math-info.univ-paris5.fr

Abstract

We present in this paper a new method of analysis and decomposition of handwritten documents into glyphs (graphemes) and their associated code book. The different techniques that are involved in this paper are inspired by image processing methods in a large sense and mathematical models implying graph coloring. Our approaches provide firstly a rapid and detailed characterization of handwritten shapes based on dynamic tracking of the handwriting (curvature, thickness, direction, etc.) and also a very efficient analysis method for the categorization of basic shapes (graphemes). The tools that we have produced enable paleographers to study quickly and more accurately a large volume of manuscripts and to extract a large number of characteristics that are specific to an individual or an era.

1. Introduction

This paper is part of the project ANR GRAPHEM[1]. It represents a methodological contribution for handwriting characterization and classification that must e used to assist the paleographers in their delicate task of writings' expertise. We are especially interested in ancient Latin manuscripts of the Middle Ages before the emergence of printing. On such manuscripts, we must face several constraints due to the bad conditions of handwriting preservation and especially of ink degradations. Furthermore, the execution rules of writing are in paleography very strict: some letters and letter combination can be only produced by a unique dynamic movement. In our study, it is necessary to take into account all these constraints and some execution specificities of writings of the Middle Age period. In this context, our objective is to:

- Produce a decomposition of writings into coherent graphemes while avoiding some backward movement of the pen.
- Produce a robust and flexible construction of Shapes Code Books that are representative to the distribution and graphemes. We will show how these elements can be used for the recognition of the different writing styles of manuscripts. We present in the following sections the weakness of conventional methods and the principle of our approach that has been designed to be adapted to ancient handwriting documents. In details, we present:
- A decomposition approach into graphemes that is based on the detection of the center line directly on the grayscale image.
- A classification approach of graphemes by graph coloring that has never been exploited in such context.

2. Decomposition of manuscripts into graphemes

This decomposition is done in two steps:
- The path tracing and medial axis detection,
- The decomposition of the extracted path into graphemes.

2.1. Path tracing and medial axis detection

2.1.1. Existing methods

Nowadays there exist a wide variety of tracing and medial axis (or skeleton) detection methods of strokes.

[1] ANR Graphem Project 2007-2010. http://liris.cnrs.fr/graphem/

In the literature, these methods can be grouped into four categories based on (for details see [1]):

- ***Morphological thinning:*** thinning is to gradually remove the points of the contour shape, while maintaining its topological features. These methods require a prior binarization step of grayscale images. This can lead to a great loss of information when the documents are ancient (with poor quality) and produces degraded binary lines, broken characters, merged or biased (holes, noise) [2] [3]. These degradations often distort the skeleton of the stroke and cause significant errors in stroke matching [4]. These limitations lead to other skeletonization methods that are applied directly on grayscale images, like the methods that are base on the 2D potential fields [5]. Such approaches require costly operations of smoothing, but they are more robust to image degradations.

- ***Distance transform:*** the distance map of an object is to associate to every point its distance to the closest edge point. The local maxima of the distance map correspond exactly to the points of the skeleton of the object. Several distances were used in this context (Euclidean [6], Chamfer [7], etc.) mostly applied on binary images rather than grayscale images.

- ***Heuristics:*** these methods are directly applied on grayscale images using heuristics to set a large number of parameters that manage the detection of the medial axis. They were originally developed to extract the skeleton of fingerprints and their results are clearly more robust on degraded image than the two previous families [3] [8].

- ***Edge detection:*** these methods use the edge to navigate along the lines and to detect the medial axis by correlation between a line and its two edges. In this context, they proposed an interesting iterative approach in [9] that detects the center line in images of neurons. Other methods that are founded on the same principle are used for road tracing in satellite images [10] [11]. This type of methods cannot be applied easily on degraded manuscripts where the contours of strokes are often distorted and discontinuous and where the contour tracing algorithm may be lost in these parasite channels.

2.1.2. Our medial axis detection method

Our approach is applied directly on grayscale images of ancient manuscripts. It combines the aspects of the three previous categories cited above offering a tracking and medial axis detection of the path that is more precise and robust to degradations for this type of documents. That is why; our method combines many concepts in the same time:

- It uses the concept of Xu et al. [12] to be able to navigate along the strokes and to extract the medial axis without using the contours. It takes into account the change of curvature of the trajectory at each point of the path where we must find the next point. This approach ensures good robustness to discontinuities encountered in the ancient manuscripts.

- In the method of Xu et al. the user must arbitrary initialize the starting points and directions. To avoid errors related to the user we use the skeleton that is obtained by the method of Zhang [13] to automatically place the starting points of the stroke that are to be traced in the neighborhood to the ends of strokes. Within this approach, we can automatically associate to detected points the radii values and the stroke directions.

-We improved the method of Xu by using at each point the combination of two complementary directions. a) Geometric direction to insure certain robustness to undesirable bifurcation and crossing situations. b) Intensity direction that guarantees a tracing that is robust against sudden direction changes or thickness along the lines. We also used a dynamic size window to find the next point that will belong to the medial axis. The size of this window varies automatically if the situation is a bifurcation, crossing or a straight line.

- The concept of Frangi [14] for the detection of blood vessels has been reused in our work. Its approach has some properties that can be compared to black lines in the manuscripts. We have been inspired by Frangi's strategy for automatic, dynamic and progressive calculation of the radius at each point of the path. This strategy allows to avoid using a radius of fixed size imposed by the method of Xu [12] (Figure 1).

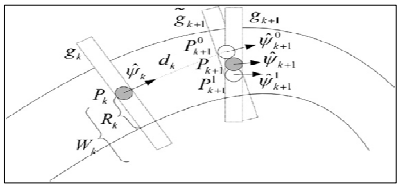


Figure 1. Line tracing and medial axis detection.

The steps of our method of tracing and detection of the medial axis are summarized in the following algorithm:

a) Initialization

- Detect the starting points by using the skeleton of Zhang,
- Extract their radius,
- Start from one of the points of departure,
- Determine the next point P^0_{k+1} by using d_k (Look ahead Distance),
- Compute its direction Ψ^0_{k+1} ,

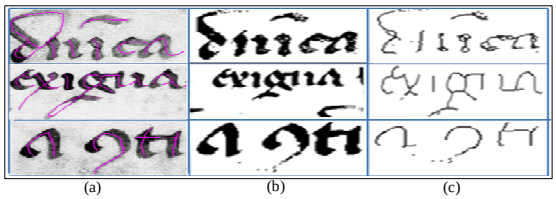
b) Determine the next point and adjust its position

- Draw a density profile g_{k+1} at P_{k+1}^0 perpendicular to its direction Ψ_{k+1}^0. Matched filtering is applied to the profile to obtain the center point P_{k+1}^0 and the tracking direction is updated to Ψ_{k+1}^1, this point will have the value of the radius that was computed previously.
- Proceed in the same way by computing the density profile at the point P_{k+1}^1, focal point in the centerline, and compute its direction Ψ_{k+1},
- Mark this point as a visited point this way it will not be visited again by our tracing algorithm.

C) Stopping Criteria

- If we find a bifurcation point and that point has already been visited, we stop the tracing process if we come at a point that is marked as starting point, and this point will be removed from the starting points list.
- Reiteration of the process along the line until a stopping criterion is met.

The following figure shows the centerline extraction results of our method. It is significantly better than that obtained by the conventional method of Zhang. We note that our method detects the medial axis even in situations where the ink is degraded or very bright.



(a) (b) (c)

Figure 2. Medial axis extraction by, (a) our method, (b) binarisation by using sauvola, (c) the method of zhang,

2.2. Path decomposition into graphemes

From a methodological point of view, the segmentation of strokes is performed as follows: between each start and stop, all the points involved in the formation of a stroke will be saved in a list with their directions and the thickness. The points of minimum thickness (local minimum) are then marked and offered as a point of cutting, as it is the case in the formation of a line, see Figure 3. In this figure, each segment has a different coloration.

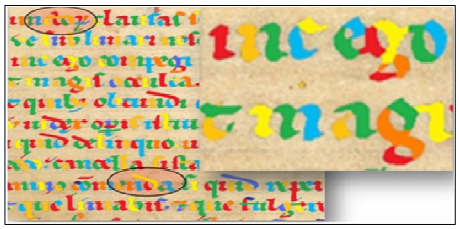


Figure 3. Example of strokes decomposition into graphemes by our method.

We identify by this approach the crossing areas, the points where the feather has been raised or where the feather has been put (see zoom in Figure 3). The decomposition of Figure 3 shows that the letters are formed from adjacent fragments attached to points of minimum thickness. These points are supposed to correspond to posing and raising points.

So as to be able to build an informative shapes codebook for each handwriting of the database, we must proceed to a graphemes classification. We have been interested in an innovative clustering approach of similar graphemes by using the concept of graph coloring that has never been exploited in such context. This method does not require prior knowledge of the number of classes or prior learning and can offer good intra class homogeneity and inter class disparity. It can adapt better adapted to the nature of the graphemes, in contrast to conventional methods which require the prior introduction of many classes and the manual labeling of graphemes. During this previous steps of learning, the user can introduce errors. We can cite for example the method of Zhu [15] and the method of Kumar [16] that are based on the SVM and the method of Schomaker[17] that is based on the Kohonen map.

3. Theoretical principle of the graph coloring

Graph coloring is a very important branch of graph theory. Its applications are numerous in various scientific fields (optimization of transportation or communication networks, chemical formulas, etc.). The definitions of graph coloring are simple and real research problems can be expressed in a well structured form. This model was first introduced in the field of documents by Gaceb and Eglin [18]. They adapted it to all the steps of documents analysis (from the physical structure extraction to the recognition) to consolidate the cooperation and exchange of information between different modules. Thanks to its simplicity and its potential for classification, we can develop a novel method for codebook construction.

3.1. Types of existing coloring methods

Graph coloring is a tool that permits graph characterization. There are many types of colorations. We can mention, for example, the coloring of nodes (the one that interests us), the coloring of edges and the list coloring [20]. The coloring of a graph *G(V,E)* is a function that affects a color to each node, and is such that two nodes that are connected by an edge (or

adjacent neighbors) do not have the same color (property constraint). The colors (or integers) assigned to the nodes of the graph are only used to group the nodes into classes.

3.2. Modeling of the grapheme classification problem in terms of graph coloring

The main objective of the combination of a set $X=\{x_1,...,x_n\}$ of graphemes into many homogeneous groups is to collect as much similar graphemes as possible, in the lowest number of classes. The grouping is based on a similarity criteria S that specifies if a pair of graphemes (x_i,x_j) can be merged into one group or not. This criterion can be based on a distance lower than a threshold. In a practical point of view, we represent each grapheme x_i by a node $v_i \in V$ of a simple graph G and we add an edge $E(v_i,v_j)$ between each pair of dissimilar graphemes (that do not respect the constraint S).

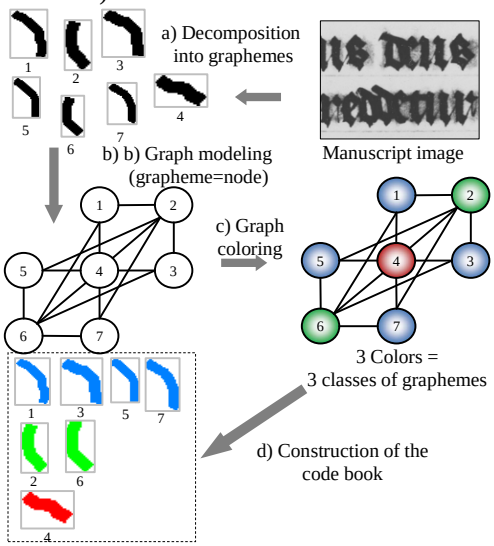


Figure 4. Construction states of the codebook by graph coloring.

The coloring of the nodes of $G(V,E)$ consists in assigning to all nodes a color so that two adjacent nodes (dissimilar) can't have the same color. These colors will correspond to different homogeneous groups that represent the different classes of graphemes. In this problem of clustering, the question of determining the smallest number of homogeneous groups is equivalent to finding the smallest k for which the corresponding graph G admits k-coloring: Thus, it is precisely the chromatic number $\chi(G)$ of the graph G that we should determine.

Compared with conventional clustering mechanisms, this modeling has the advantage to easily manage several kinds of ambiguities related to the shape of graphemes.

4. Construction of the code books

From all graphemes extracted in section 2, we will proceed with the construction of our code book (also called similarity table).

4.1. Features extraction

Thus we produce a vectorial description of each of the graphemes, which is then processed to define the similarity criterias that are necessary for the classification. We used according to need two types of descriptors: a topological descriptor of 15 features and a descriptor of 25 Zernike Moments.

4.1.1. Topological features

- 10 shape features were extracted from the binary grapheme images. The length and width of the grapheme are used to distinguish between the writing styles and also between the used feathers. The orientation allows to know the inclination of the grapheme and to differentiate between the various executions movements of writing. The thickness of the grapheme allows us to know the thickness of the head of the feather and the writing style. For example, in Gothic manuscripts, the thickness of the head of the feather is large while in the modern manuscripts it is small. The last three features are related to the surface that a grapheme can take in a manuscript, compared to the other graphemes.

- 5 curvature features (the direction of the greatest curvature and of the least curvature, the Gaussian and mean curvatures and the laplacian of the curve) are extracted from the grayscale graphemes and are calculated by using the Hessian Matrix [14]. They represent the shape of the curves of graphemes and reflect the structural properties such as convexity and concavity. They have an important role, because they allow us to differentiate between the style and the era of the writing.

4.1.2. Zernike Moments

The 25 Zernike moments used to describe the graphemes are classified as orthogonal moments (geometric, Legendre, etc...), because they possess the property of invariance to rotation. The Zernike moment

of order n with the repetition m ($n-|m|$ is even and $|m|<n$) is defined by:

$$A_{nm} = \frac{n+1}{\pi} \sum_x \sum_y f(x,y) \cdot V^*_{nm}(\rho,\theta)$$

$V_{nm}(\rho,\theta)$Is a set of complex polynomials in a two dimensional space that form an orthogonal set on the interior of the unit circle ($x^2+y^2=1$), with :

$$V_{nm}(\rho,\theta) = R_{nm}\, e^{im\theta}$$

Where ρ is the length of the origin vector at the point with coordinates (x,y). θ Is the angle between the vector ρ and the x-axis. $R_{n,m}=R_{n,-m}$ is a radial polynom defined as follows:

$$R_{nm}(\rho) = \sum_{S=0}^{(n-|m|)/2} (-1)^S \frac{(n-S).!}{S!\left(\frac{n+|m|}{2}-S\right)!\left(\frac{n-|m|}{2}-S\right)!} \rho^{n-2S}$$

To calculate the Zernike moments, the center of the binary grapheme is taken as the origin of the coordinate system and pixel coordinates of the image are transformed in a way to be in the domain of the unit circle. As mentioned earlier, the Zernike moments are only invariant to rotation, to become invariant to scaling [19] we must normalize the binary image of grapheme by the first order moment m_{00} defined as being the area of the grapheme.

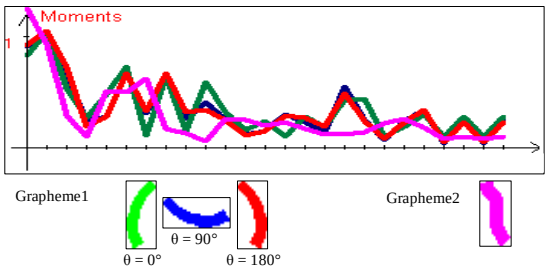


Figure 5. Invariance to rotation of the 25 Zernike moments (n = 8).

4.2. Similarity Measure

The dissimilarity between v_i and v_j is given by the generalized Minkowski distance of order α ($\alpha=2$: euclidean distance). $Ds = \left(\sum_{k=1}^{Nc} g_k(v_i^k, v_j^k)^\alpha \right)^{\frac{1}{\alpha}}$

Nc is the length of the feature vector. g_k is the dissimilarity function that compares the features in pairs.

4.3. Graph Construction

The construction of a graph G to be colored from a set $X=\{x_1,\ldots, x_n\}$ of n graphemes (where each node vi corresponds to the descriptor vector of the grapheme x_i) and is mainly based on the computation of the matrix of distances MD_s that reflect the dissimilarities $Ds(x_i, x_j)$ existing between the pairs of graphemes (x_i,x_j) given by the following equation : $MDs[v_i,v_j] = Ds(x_i, x_j)$ with $i \in [1,n]$ et $j \in [1,n] \mid (i \neq j)$.

Once MDs is calculated, we associate to X a graph with a higher threshold $G_{\geq S} = (V=X, E_{\geq S})$ by using the following equation:

$$E_{\geq S}[v_i, v_j] = \begin{cases} 1 & \text{si } Ds(x_i,x_j) = Ds(v_i,v_j) \geq S \\ 0 & \text{sinon} \end{cases}$$

For not to confuse the term adjacency with the term similarity, it should be noted that two nodes are adjacent if they have a higher dissimilarity threshold S. The threshold S is also called adjacency threshold.
This threshold can be adjusted manually by the paleographers or automatically by maximizing the quality of the classification given by ψ [18]:

$$S^{Optimal} = \arg\max(\psi(S_i))$$

4.4. Graphemes classification

Once the graph G is constructed from the set of grapheme, we apply the graph coloring algorithm of Gaceb and Eglin [18]. The resulting different colors represent the classes of the graphemes.

5. Results and application

We present here our grapheme decomposition method on 12 pages of ancient manuscripts from different types.

Figure 6. Samples of the 12 manuscripts.

The following set of 4863 graphemes summarizes the number of graphemes obtained by each page: *{p1=343, p2=583, p3=643, p4=248, p5=398, p6=528, p7=564, p8=316, p9=499, p10=193, p11=269, p12=279}.*
This decomposition has been subject to the validation of the paleographers and has obtained their approval.

The following figure shows a sample of the code-book that was constructed from the graphemes of page 12 by graph coloring (see Section 4). It is quite clear

that the graphemes that represent the same motion of the feather are grouped in one class.

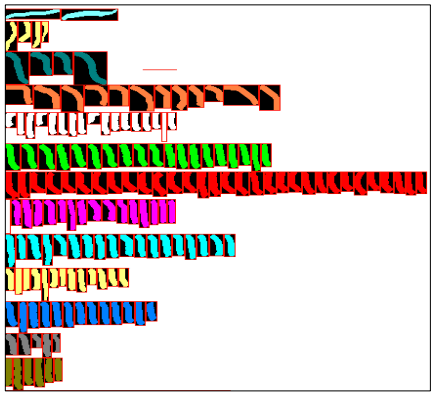


Figure 7. A Sample of page's 12 codebook.

To recognize the style of the manuscripts: each of the 12 manuscript pages pi will be represented by its codebook $cb(p_i)$ of k_i classes of graphemes (k_i is the chromatic number of the graph coloring) with $cb(p_i) = \{c_1^i, ..., c_k^i\}$ Where each class c_t contains n_t graphemes with $c_t = \{x_1^t, ..., x_{nt}^t\}$. Thus we can estimate the similarity between each pair of pages (p_i, p_j) by the following distance dp:

$$dp\left[cb(p_i), cb(p_j)\right] = \sup\left[dc(c_n, c_m)\right] | n = 1..k_i \text{ et } m = 1..k_j$$

Where dc is the distance between two classes of graphemes and it is given by:

$$dc(c_n, c_m) = \min\{Ds(x_i \in c_n, x_j \in c_m)\}$$

The distances dp between the twelve pages are illustrated table1. We can estimate from table 2 that the style of page 7 is the closest to the one of page 1 and the style of page 2 is the closest to the style of page 10 and so on.

Table 1. The distances between the codebooks of the 12 pages.

	1	2	3	4	5	6	7	8	9	10	11	12
1	0											
2	0.7145	0										
3	0.391	0.459	0									
4	0.3063	0.679	0.3464	0								
5	0.1559	0.596	0.3353	0.2483	0							
6	0.5464	0.391	0.7943	0.5372	0.4591	0						
7	0.127	0.422	0.3173	0.3419	0.1755	0.3867	0					
8	0.4871	0.448	0.8065	0.5029	0.5351	0.5487	0.2362	0				
9	0.7056	0.231	0.8526	0.7685	0.6276	0.4637	0.4141	0.7529	0			
10	0.8062	0.206	0.728	0.6466	0.6261	0.4637	0.1435	0.7122	0.153	0		
11	0.3437	0.597	0.2586	0.1511	0.2446	0.1863	0.208	0.3852	0.555	0.5772	0	
12	0.3413	0.603	0.3659	0.1331	0.2381	0.3868	0.1741	0.3883	0.589	0.6152	0.2295	0

6. Conclusion

We have presented in this article new methods for handwriting decomposition into graphemes and a robust codebook that is well adapted for the demands of our field. Applying these different methods, we succeed to automatically recognize the style and era of the manuscripts presented in this set of test.

References

[1] H. Daher et al. A New approach for centerline extraction in handwritten strokes…, *International Workshop on Document Analysis Systems*, Boston, 2010.

[2] D. Lee, S.W. Lee. A new methodology for gray-scale character segmentation and recognition, *ICDAR*, vol. 1, pp.524, 1995.

[3] D. Maio, D. Maltoni, Direct Gray-Scale Minutiae Detection In Fingerprints, *IEEE Transactions on PAMI*, vol. 19, n°. 1, pp. 27-40, 1997.

[4] Suh et al. Stroke extraction from gray-scale character image, *Progress in Handwriting Recognition* 593-598, 1997.

[5] T. Grigorishin et al. Skeletonisation: An Electrostatic Field Based Approach, *Pattern Analysis & Applications*, vol. 1, pp. 163-177, 1998.

[6] P.E. Danielsson. Euclidean Distance Mapping, *Computer Graphics and Image Processing*, vol. 14, pp. 227-248, 1980.

[7] A. Rosenfeld and J.L. Pfalz. Distance Functions on Digital Pictures, *Pattern Recognition*, vol. 1, pp. 33-61, 1968.

[8] Yaxuan Qi. Fingerprint Ridge Line Reconstruction. *Intelligent Information Processing*, pp. 211-220, 2004.

[9] Y. Zhang et al. A novel tracing algorithm for high throughout imaging Screening of neuron-based assays, *J Neurosci Methods* 160, pp. 149–162, 2007.

[10] D.Poz et al. Automated extraction of road network from medium and high-resolution images, *Pattern Recognition and Image Analysis*, vol. 16, n°. 2, pp 239-248, 2006.

[11] R. Peteri et al. Detection and extraction of road networks from high resolution satellite images, *International Conference on Image Processing*, vol.1, pp I-301-4, 2003.

[12] Y. Xu et al. An improved algorithm for vessel centerline tracking in coronary angiograms, *Computer Methods and Programs in Biomedicine*, Vol. 88, n° 2, Pages 131-143.

[13] T.Y. ZHANG et C.Y. SUEN : A fast parallel algorithm for thinning digital patterns. *Communications of the ACM*, 27(3):236–240, mars 1984.

[14] A.F. Frangi et al. Multiscale Vessel Enhancement Filtering, MICCAI '98, pp. 130-137, 1998.

[15] G. Zhu et al. Language Identification for Handwritten Document Images Using A Shape Codebook.

[16] J. Kumar et al. Handwritten Arabic Text Zone Detection using A Shape Codebook. ICPR, 2010.

[17] L. Schomaker et al. Using codebooks of fragmented connected-component contours in forensic and historic writer identification. *Pattern Recognition Letters* 28(6), pp 719-727, 2007.

[18] D. Gaceb et V. Eglin : Improvement of postal mail sorting system. *IJDAR*, 11(2):67-80, 2008.

[19] M.R.Teague, Image analysis via the general theory of moments , *J.opt.soc.Am*, vol.70, n°8, pp 920-930, 1980.

[20] V. PASCHOS, book, Optimisation combinatoire5: problèmes paradigmatiques et nouvelles problématiques, *Lavoisier*, France, pp. 270, 2007.